**Задача 1**

Используя docker поднимите инстанс MySQL (версию 8). Данные БД сохраните в volume.

Изучите [бэкап БД](https://github.com/netology-code/virt-homeworks/tree/master/06-db-03-mysql/test_data) и восстановитесь из него.

Перейдите в управляющую консоль mysql внутри контейнера.

root@ubuntusql-VirtualBox:~# docker run -ti -d --name mysqldb -p 3306:3306 -e MYSQL\_ROOT\_PASSWORD=password -v my-datavolume:/var/lib/mysql mysql:8.0 # запустить образ с volume mysql

root@ubuntusql-VirtualBox:/home/ubuntusql/sql# docker exec -ti df50840f7d67 bash # попадаем в контейнер

root@8aea5f5583ff:/var/lib/mysql# curl -LO <https://raw.githubusercontent.com/netology-code/virt-homeworks/master/06-db-03-mysql/test_data/test_dump.sql> # скачиваем курлом файлик где

***-L принимать и обрабатывать перенаправления;  
-O - сохранять контент в файл с именем страницы или файла на сервере;***

root@8aea5f5583ff:/var/lib/mysql# mysql -uroot –p # залетаем в mysql

mysql> create database test\_db;

mysql> use test\_db;

mysql> source test\_dump.sql

mysql> \s

--------------

mysql Ver 8.0.25 for Linux on x86\_64 (MySQL Community Server - GPL)

Connection id: 34

Current database: test\_db

Current user: root@localhost

SSL: Not in use

Current pager: stdout

Using outfile: ''

Using delimiter: ;

Server version: 8.0.25 MySQL Community Server - GPL

Protocol version: 10

Connection: Localhost via UNIX socket

Server characterset: utf8mb4

Db characterset: utf8mb4

Client characterset: latin1

Conn. characterset: latin1

UNIX socket: /var/run/mysqld/mysqld.sock

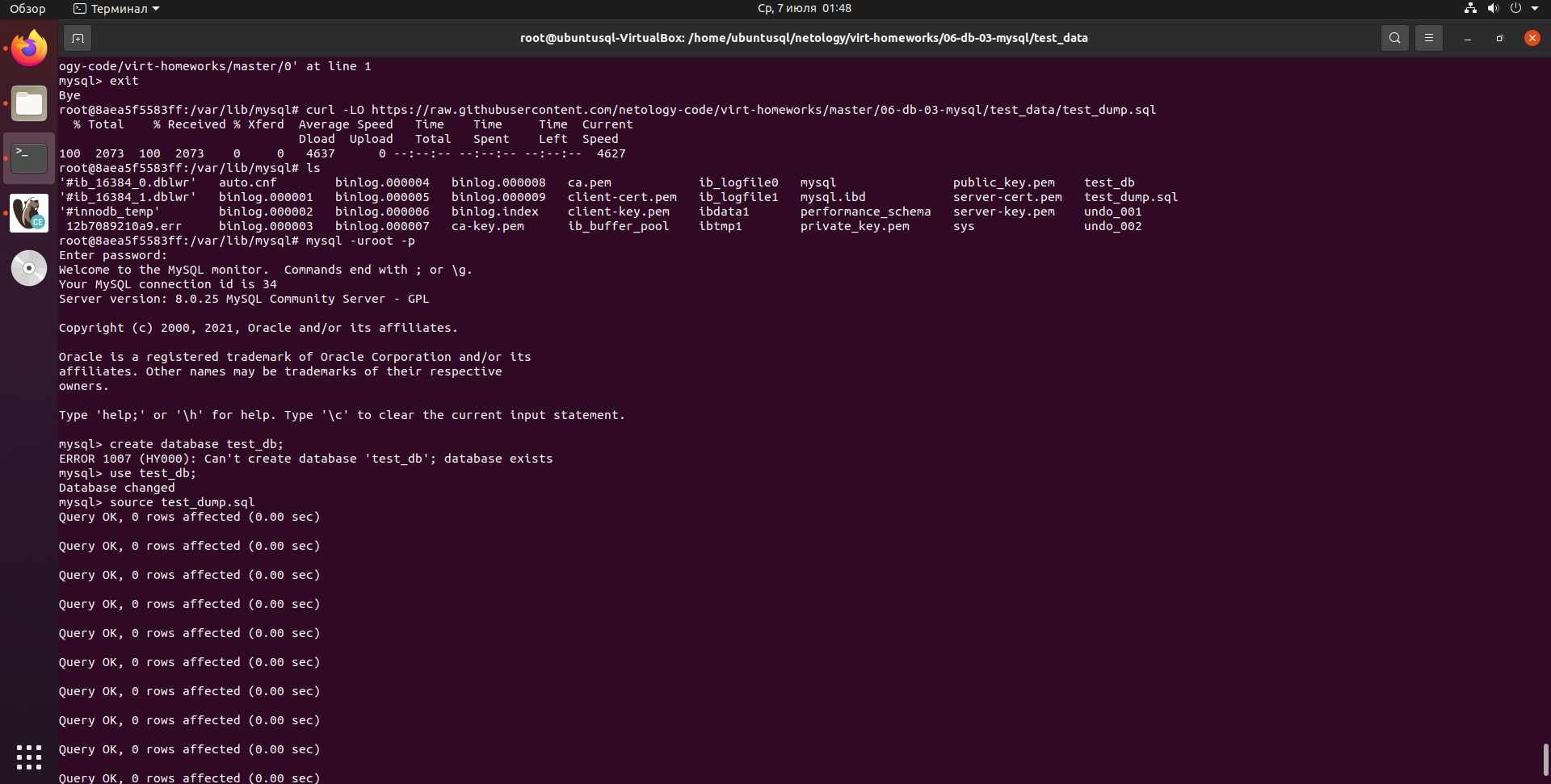
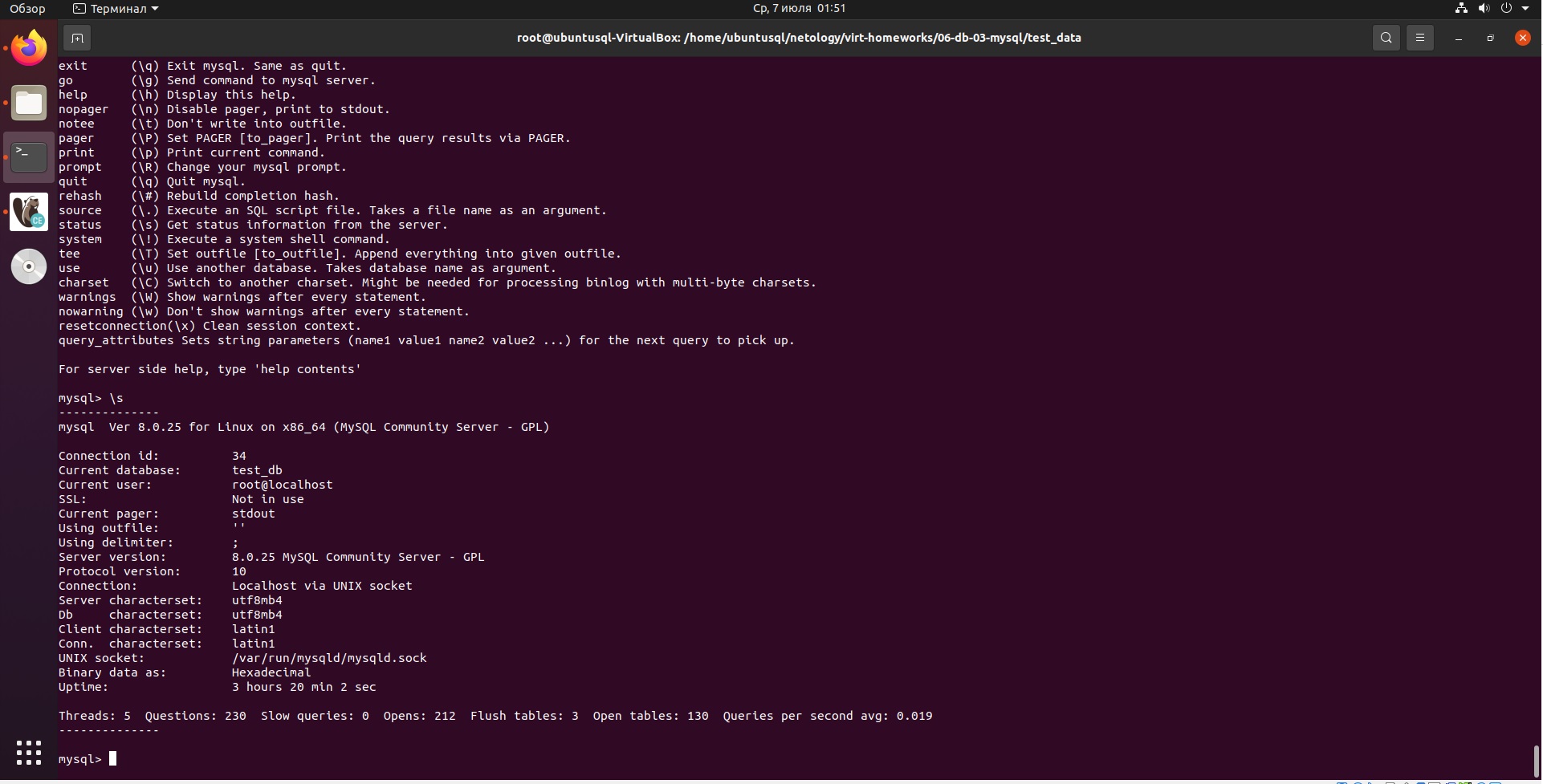
Binary data as: Hexadecimal

Uptime: 3 hours 20 min 2 sec

Threads: 5 Questions: 230 Slow queries: 0 Opens: 212 Flush tables: 3 Open tables: 130 Queries per second avg: 0.019

--------------

см. скрин “Восстановили dump Домашнее задание к занятию 6.3. MySQL.jpg”



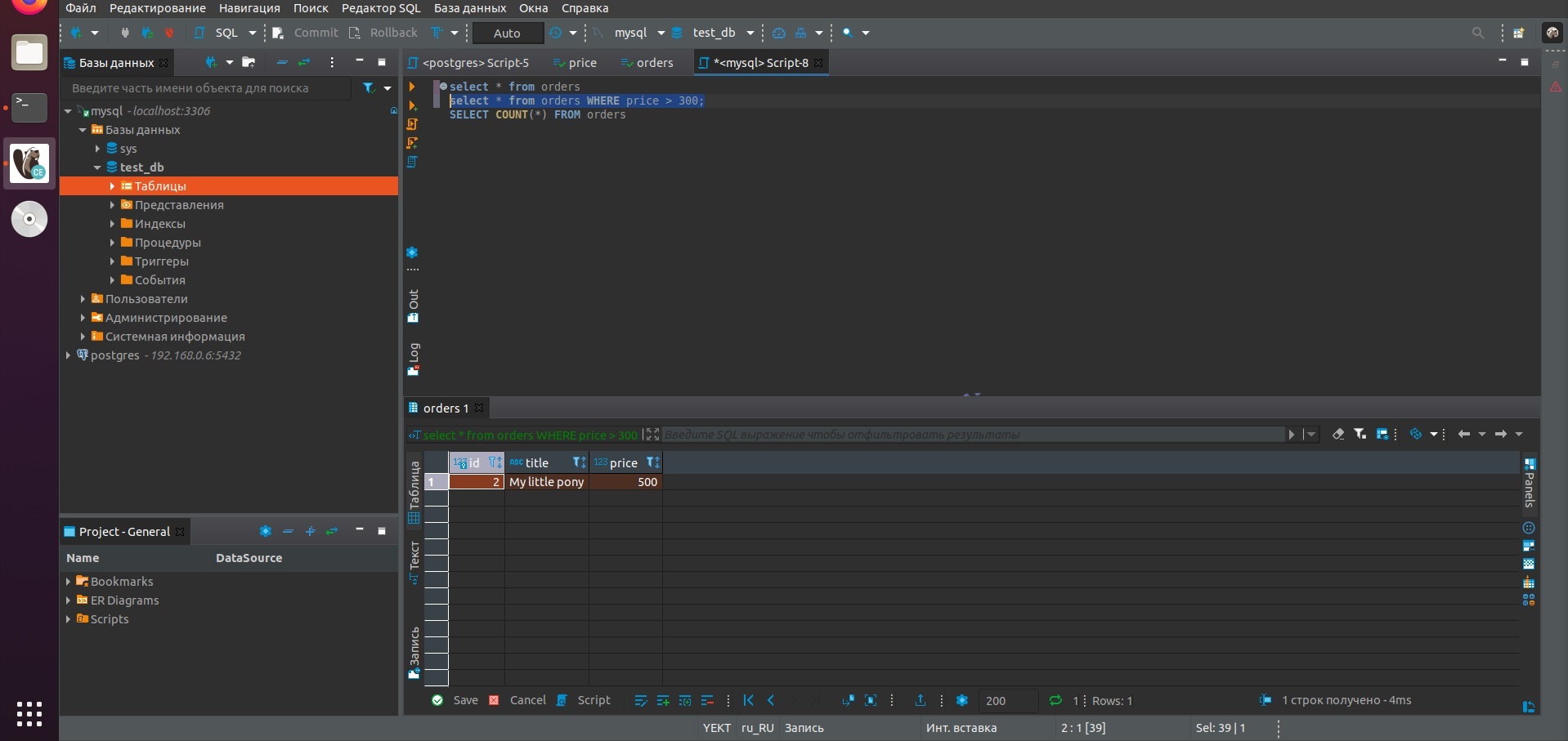
Используя команду \h получите список управляющих команд.

**Приведите в ответе** количество записей с price > 300.

select \* from orders WHERE price > 300;

SELECT COUNT(\*) FROM orders WHERE price > 300;

См. скрин “Количество записей.jpg”



**Задача 2**

Создайте пользователя test в БД c паролем test-pass, используя:

* плагин авторизации mysql\_native\_password
* срок истечения пароля - 180 дней
* количество попыток авторизации - 3
* максимальное количество запросов в час - 100
* аттрибуты пользователя:
  + Фамилия "Pretty"
  + Имя "James"

Предоставьте привелегии пользователю test на операции SELECT базы test\_db.

Используя таблицу INFORMATION\_SCHEMA.USER\_ATTRIBUTES получите данные по пользователю test и **приведите в ответе к задаче**.

CREATE USER 'test'@'%' IDENTIFIED BY 'test-pass'

ATTRIBUTE '{"Family": "Pretty", "name": "James"}'; # создаем пользователя с паролем test-pass и присваиваем атрибуты фамилию и имя.

ALTER USER 'test'@'%' PASSWORD EXPIRE INTERVAL 180 DAY

FAILED\_LOGIN\_ATTEMPTS 3 PASSWORD\_LOCK\_TIME 2;

# делаем срок истечения пароля ## каждый 180 дней придется менять  
# и включить отслеживание неудачных попыток входа в систему, чтобы три последовательных неверных пароля вызывали временную блокировку учетной записи на два дня

ALTER USER 'test'@'%'

WITH

MAX\_QUERIES\_PER\_HOUR 100;

# максимальное количество запросов в час

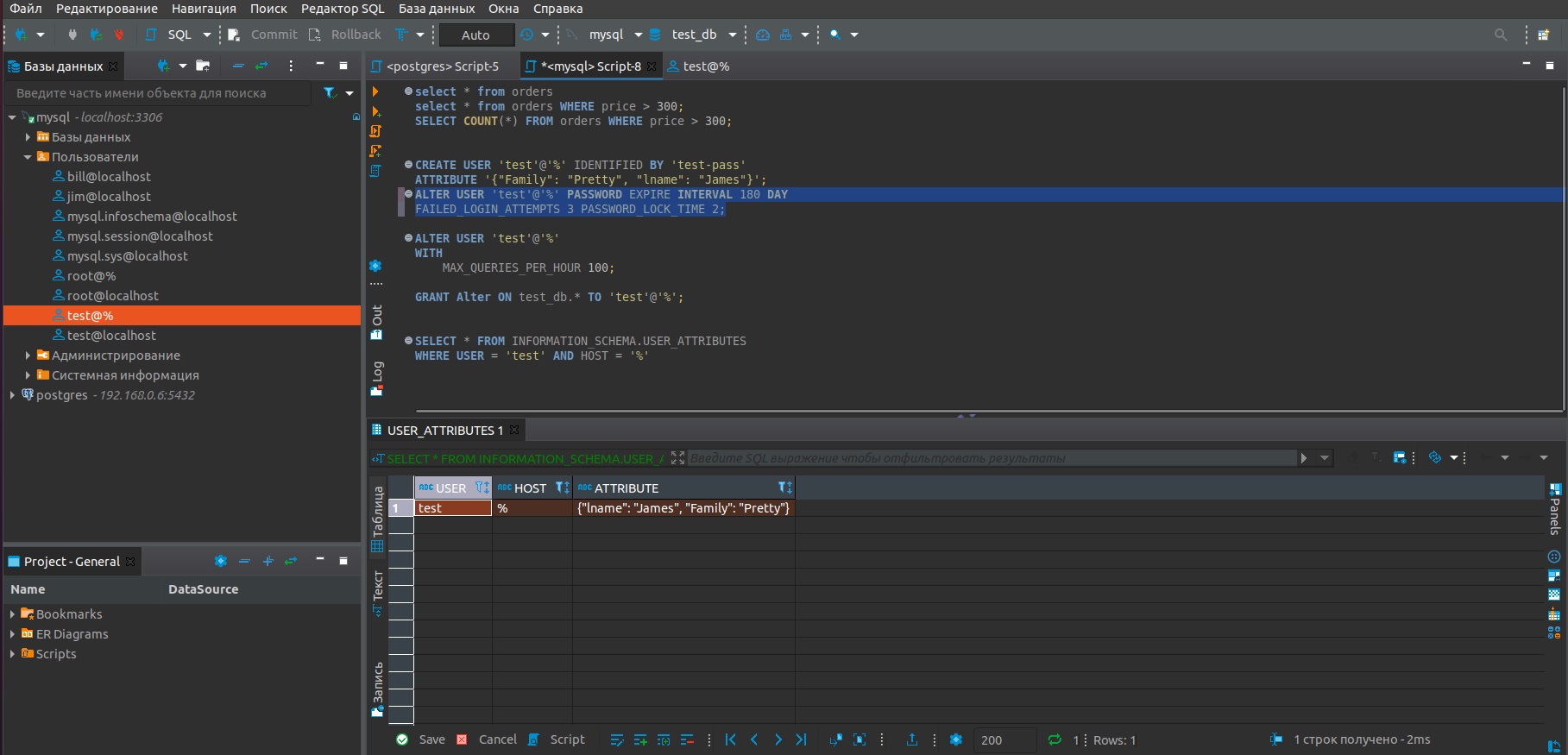
GRANT Alter ON test\_db.\* TO 'test'@'%'; # наделяем правами

Используя таблицу INFORMATION\_SCHEMA.USER\_ATTRIBUTES получите данные по пользователю test и **приведите в ответе к задаче**.

SELECT \* FROM INFORMATION\_SCHEMA.USER\_ATTRIBUTES

WHERE USER = 'test' AND HOST = '%' # получите данные по пользователю test

См. скрин “INFORMATION\_SCHEMA.USER\_ATTRIBUTES Домашнее задание к занятию 6.3. MySQL.jpg”



**Задача 3**

Установите профилирование SET profiling = 1. Изучите вывод профилирования команд SHOW PROFILES;.

Исследуйте, какой engine используется в таблице БД test\_db и **приведите в ответе**.

Измените engine и **приведите время выполнения и запрос на изменения из профайлера в ответе**:

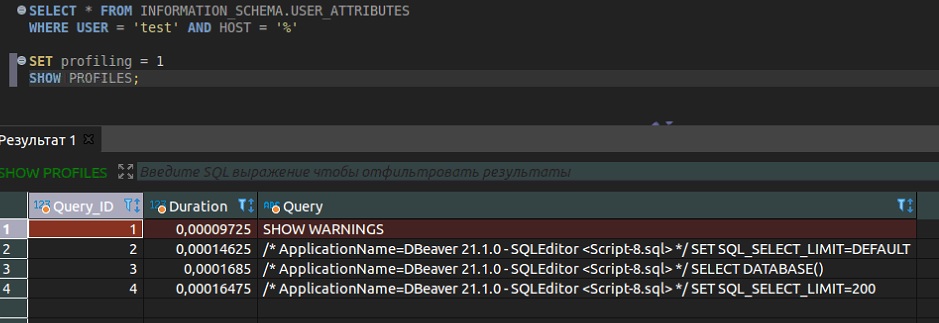
* на MyISAM
* на InnoDB

Ответ:

SET profiling = 1# включитьпрофилирование

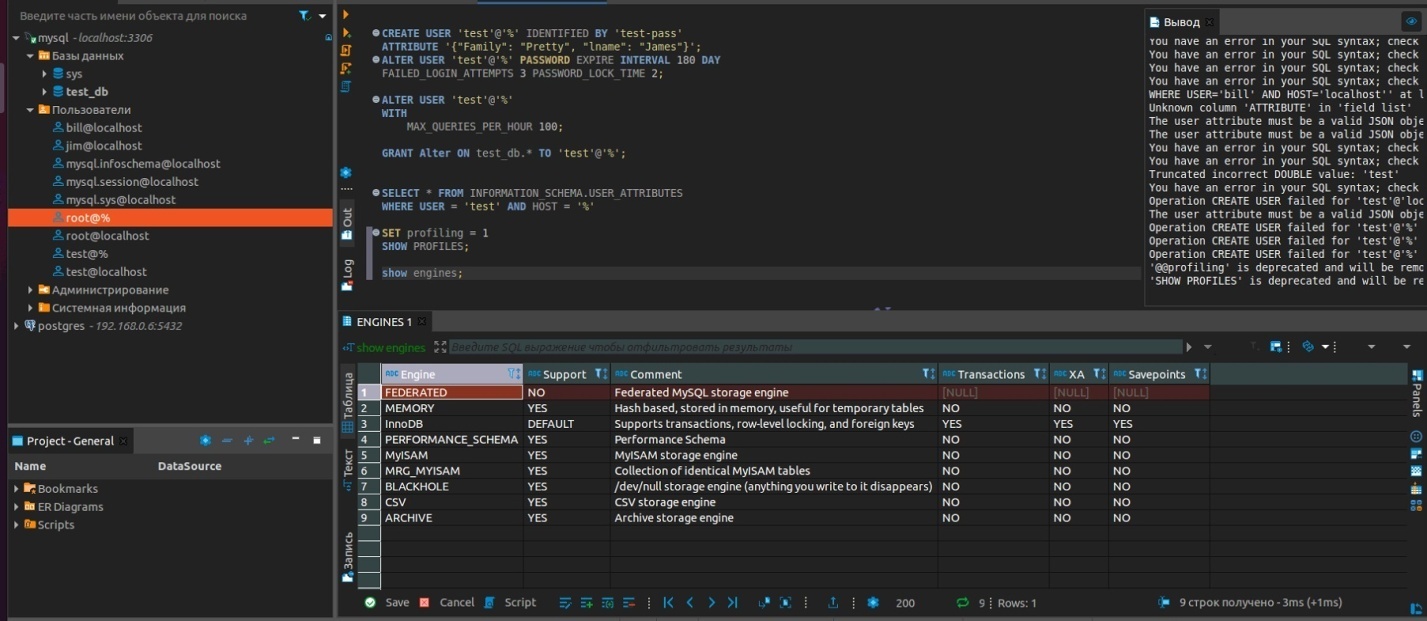
SHOW PROFILES;# покажипрофилирование

См.Скрин“profiling Домашнее задание к занятию 6.3. MySQL.jpg”



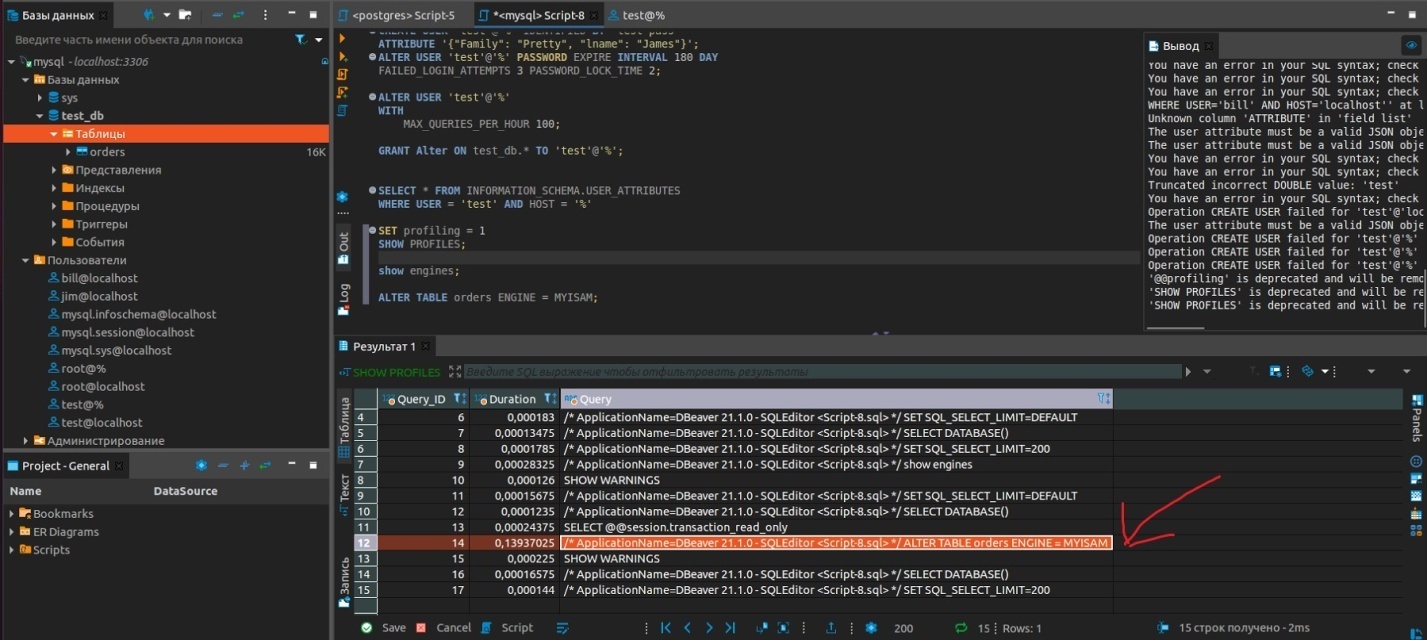
show engines; # покажи движки включая по умолчанию

см. скрин“show\_engines\_default\_Домашнее\_задание\_к\_занятию\_6\_3\_MySQL”



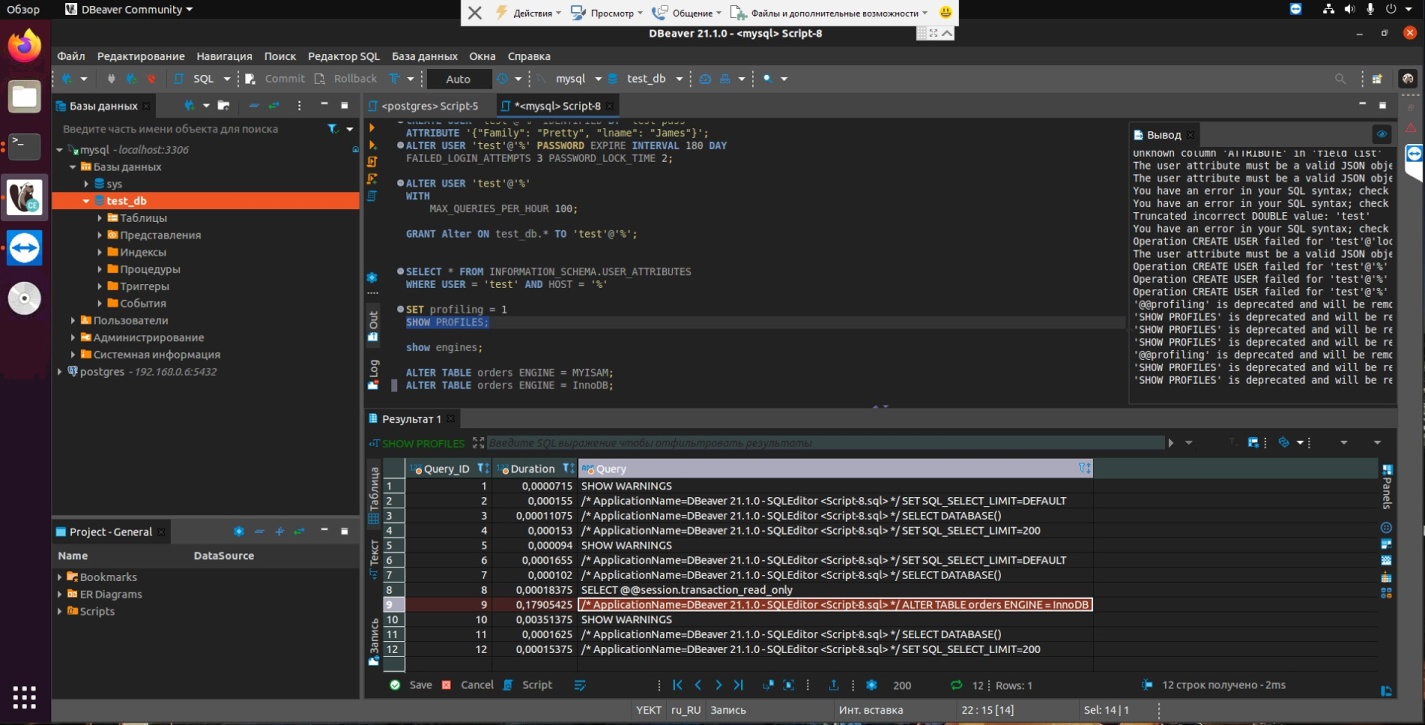
ALTER TABLE orders ENGINE = MYISAM; # Смени таблицу orders движок на MYISAM

См. скрин “смена\_движка\_на\_MYISAM\_Домашнее\_задание\_к\_занятию\_6\_3\_MySQL.jgp”



ALTER TABLE orders ENGINE = InnoDB; # смени таблицу orders движок на InnoDB

См. скрин “смена\_движка\_на\_InnoDB\_Домашнее\_задание\_к\_занятию\_6\_3\_MySQL.jpg”



**Задача 4**

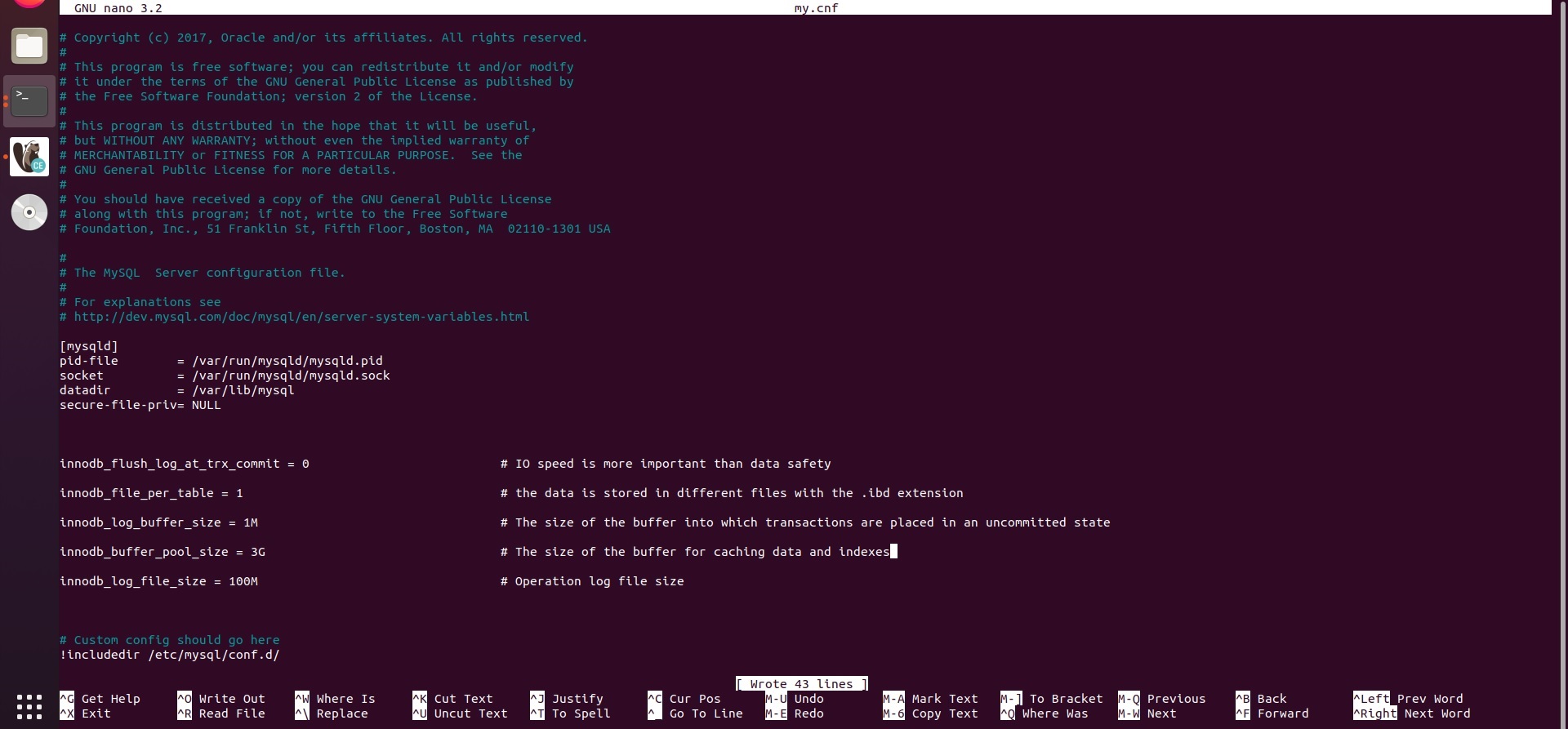
Изучите файл my.cnf в директории /etc/mysql.

Измените его согласно ТЗ (движок InnoDB):

* Скорость IO важнее сохранности данных
* Нужна компрессия таблиц для экономии места на диске
* Размер буффера с незакомиченными транзакциями 1 Мб
* Буффер кеширования 30% от ОЗУ
* Размер файла логов операций 100 Мб

Приведите в ответе измененный файл my.cnf.

См. скрин “My-cnf Домашнее задание к занятию 6.3. MySQL.jpg”



innodb\_flush\_log\_at\_trx\_commit = 0 # IO speed is more important than data safety

innodb\_file\_per\_table = 1 # the data is stored in different files with the .ibd extension

innodb\_log\_buffer\_size = 1M # The size of the buffer into which transactions are placed in an uncommitted state

innodb\_buffer\_pool\_size = 3G # The size of the buffer for caching data and indexes

innodb\_log\_file\_size = 100M # Operation log file size